



(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
27.02.2002 Bulletin 2002/09

(51) Int Cl.7: G06F 17/30

(21) Application number: 00117311.1

(22) Date of filing: 18.08.2000

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(72) Inventors:
• Heilig, Jörg
22145 Hamburg (DE)
• Laux, Thorsten O.
20099 Hamburg (DE)

(71) Applicant: SUN MICROSYSTEMS, INC.
Palo Alto, California 94303 (US)

(74) Representative: HOFFMANN - EITLE
Patent- und Rechtsanwälte Arabellastrasse 4
81925 München (DE)

(54) Data access system and method with proxy and remote processing

(57) Access system and method for accessing data in a network allowing to reduce communication load in a client and server scenario. Requests from the client may be not directly executed, but handled by a proxy server which retrieves the requested data from an appropriate data server and determines whether further

rendering or processing of the data is necessary. The data may be temporarily stored and a processing server may render the requested data appropriately, e.g., on request by the client upon receiving a link message from the proxy server, and transmit the rendered data to the client for further handling or visualization.

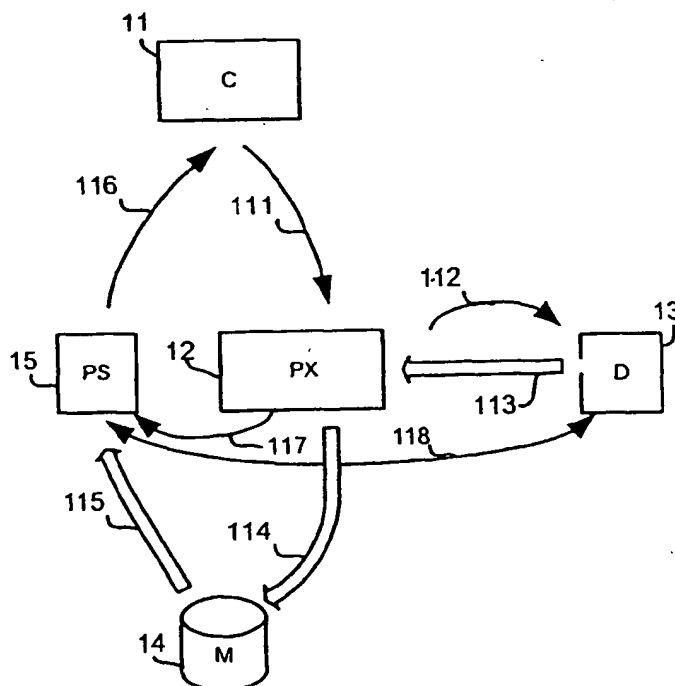


Fig. 1

BEST AVAILABLE COPY

transmitted to the client.

[0014] In a further embodiment the requested data may be retrieved from a data server and immediately stored in a memory if it is determined that the requested data have to be rendered before transmission to the client, and the processing server may be further arranged for retrieving the data stored in the memory.

[0015] The proxy server may be arranged to transmit address information of the requested data to the processing server, and the processing server may be arranged to retrieve the requested data from a data server using the address information.

[0016] The proxy server may include means or a code section containing instructions for generating a link message containing address information of the requested data, and means for transmitting the link message to the client enabling the client to request the data from the processing server.

[0017] The link message may further include at least one of the group consisting of address information of the processing server, data type information describing the requested data, a client identifier, and a session identifier.

[0018] The address information of the requested data may be constituted by an URL (uniform resource locator) and the data type information by a MIME (multipurpose internet mail extension) type.

[0019] The proxy server may include means or a code section containing instructions for direct transmission of the requested data to the client in case it is determined by the determining means that the requested data do not have to be rendered before transmission to the client.

[0020] At least the proxy server, the processing server and the temporary memory means may be connected by a local area network, by a wide area network or a combination thereof.

[0021] Further, a plurality of processing servers may be provided and means to determine at least one of the processing servers for rendering, and wherein the proxy server may be arranged to receive requests for data from a plurality of clients and to retrieve data from a plurality of data servers.

[0022] The processing server may be arranged to directly transmit the rendered data to the client or on a return path including the proxy server.

[0023] The processing server and the proxy server may be constituted by a single data processing device.

[0024] A client may comprise means or a code section containing instructions to receive and analyse a link message from an access system, and a data handler or a code section containing instructions for establishing a communication link between the client and the processing server and for receiving and handling rendered data from the processing server.

[0025] A client may comprise pre-selection means or a code section containing instructions for performing a pre-selection of requests for data into requests requiring

rendering of data and requests which do not require rendering of data and for transmitting requests requiring rendering to the proxy server, and for directly retrieving data in case it is determined that the requested data do not require rendering before transmission to the client.

[0026] Further advantageous embodiments of the invention are disclosed in further claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0027]

Fig. 1 shows a block diagram illustrating an access system according to an embodiment of the invention,

Fig. 2 shows a flow diagram illustrating a sequence of messages transmitted in the method according to another embodiment of the invention,

Fig. 3 shows a flow diagram illustrating a sequence of messages in accordance with another embodiment of the invention,

Fig. 4 shows a block diagram illustrating an access system according to an embodiment of the invention,

Fig. 5 shows a block diagram illustrating an access system according to another embodiment of the invention,

Fig. 6 shows a flow diagram illustrating a sequence of messages transmitted in accordance with another embodiment of the invention,

Fig. 7 shows a flow diagram illustrating a sequence of messages transmitted in accordance with another embodiment of the invention,

Fig. 8 shows a flow diagram illustrating a sequence of messages transmitted in accordance with another embodiment of the invention, and

Fig. 9 shows a block diagram illustrating an access system according to another embodiment of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0028] Throughout the figures corresponding reference signs denote corresponding entities.

[0029] In the following a first embodiment of the invention will be described with respect to Fig. 1. Fig. 1 shows a schematic block diagram of an access system and a client for accessing data in a network according

[0044] It is noted that the processing server 15 may render the requested data interactively under control of the client, based on instructions from the client e.g., in case a user wishes to scroll through a document or edit parts of a document.

[0045] Data transmission between the individual entities of the system shown in Fig. 1 may be accomplished via networks or dedicated communication links including wireless transmissions.

[0046] Further, it is noted that in case the proxy server 12, e.g. due to a previous request, already stored requested data in the temporary memory means 14, repeated retrieval of the same data files from a data server may be avoided by suitable measures, e.g., keeping a log of retrieved data files at the proxy server or by checking the temporary memory means upon receiving a request.

[0047] It is further possible that the access system for accessing data in a network comprises a proxy server 12 arranged for receiving a request for data from a client 11 and for determining whether the requested data have to be rendered before transmission to the client, and a processing server 15 arranged for rendering the data and for transmitting the rendered data to the client 11.

[0048] Since proxy server retrieves requested data on behalf of the client and the processing server renders the requested data, it can be avoided to transmit large amounts of data containing the originally requested data via communication links to and from the client. Instead only rendering results are transmitted to the client, potentially consisting of small amounts of data, e.g. screen contents for visualisation at the client and similar. Thus latency for accessing data from the client can be reduced, particularly in case the client communicates via low bandwidth connections, such as to and from a mobile terminal. Thus, the invention according to the above embodiment allows to reduce band width requirements.

[0049] Still further, in case it can be determined whether rendering is necessary without retrieving the requested data, e.g. at the proxy server or the client, it is possible that the requested data are not at all or not fully retrieved by the proxy server, but only information on the location of the requested data is transmitted to the processing server 15, for example a URL, as indicated by an arrow 117. The processing server can then retrieve the requested data from the data server in preparation of rendering, as indicated by an arrow 118.

[0050] In this case the proxy server may retrieve some of the requested data, for example a part of the requested data including data type information until a decision on rendering is possible and then stop retrieving the requested data. The already retrieved portion of the data may then be discarded. The temporary memory means 14 is not necessary in this case.

[0051] It is also possible that the client, upon generating a request, directly accesses the processing server, for instructing the processing server to retrieve and render the requested data.

[0052] It is noted that a computer readable medium may be provided, having a program recorded thereon, where the program is to make a computer or system of data processing devices execute functions of the above described elements, particularly of the proxy server and the processing server. A computer readable medium can be a magnetic or optical or other tangible medium on which a program is recorded, but can also be a signal, e.g., analog or digital, electromagnetic or optical, in which the program is embodied for transmission.

[0053] Further, a computer program product may be provided comprising the computer readable medium.

[0054] The functions of the processing server and the proxy server may be realized by executing code sections on a single data processing device. This data processing device may execute still further functions, such as functions of a web server or similar.

[0055] In the following a further embodiment of the invention will be described with respect to Fig. 2. Fig. 2 shows a flow diagram of steps performed in the method according to an embodiment of the invention, e.g. in case a client wishes to access data in a network.

[0056] The processing steps may be performed at the embodiment of the access system described with respect to Fig. 1, however are not limited thereto.

[0057] In a first step S21 a client generates a client request for data and transmits this request to a proxy server. The client may be configured to transmit all requests to the proxy server or only selected requests, e.g. selected in accordance with a type of request, type of data requested, data file sizes or similar.

[0058] In a step S22 the proxy server retrieves the requested data from a data server which may be accessible though a network of data processing devices such as a local area network or a wide area network. It is noted that the proxy server may omit the step of retrieving the requested data, in case the data were already retrieved at an earlier point in time and are locally available.

[0059] Then it is determined in a step S23, for example at the proxy server, whether a rendering of the retrieved data prior to transmission to the client is required. In case the decision is "YES", i.e. it is determined that rendering is necessary, the data are rendered in a step S24, for example by a processing server or by the proxy server. The rendering may be performed as outlined with respect to Fig. 1.

[0060] After performing the rendering operations rendered data are transmitted to the client in a step S25.

[0061] In case the decision in step S23 is "NO", i.e. it is determined that rendering is not necessary, the data are directly transmitted from the proxy server to the client.

[0062] It is noted that the sequence of steps may be varied, e.g. step S23 may be executed before step S22, as it may already be determined whether rendering is necessary before the requested data are retrieved.

[0063] According to the above processing steps, in

request from a client entity is routed through the proxy entity, which retrieves the requested information and transmits it to the client entity. However, it is also possible that the client 41 only sends selected requests to the proxy server 42, i.e., that the proxy server is registered at the client as a proxy only for selected requests, e.g., requests concerning certain documents, applications or data types or requests that require rendering, whereas all other requests may be directly executed by the client 41 without being routed through the proxy server, as indicated by a dashed line 401. Registration as a proxy may depend on the site to be accessed, i.e. be domain dependent. Thus Domains known to contain data requiring rendering would be accessed through the proxy server, whereas other domains could be accessed directly.

[0080] The request for data from the client 41 may be in a standard format, e.g. as used in packet switched networks. A request may, for example include an URL of a desired document.

[0081] The proxy server 42 in the embodiment of Fig. 4 comprises retrieving means 421, e.g. a retriever or a code section containing instructions for retrieving the requested data from data server 13 upon receiving the request for data from the client 41. Preferably this involves sending a request from the proxy server 42 to the data server 13. The data server then obtains the requested data from its memory and transfers the requested data back to the proxy server 42. As before, the data server 13 may be any server of a public network, such as the Internet, or may be a server of a local area network, such as a company-wide intranet. The retrieving means may also be constituted by a programmed data processing unit or realized in hardware.

[0082] Determining means 422 at the proxy server are provided for determining whether the requested data have to be rendered before transmission to the client. The determining means 422 may for example be realized by a calculation unit or by a code section containing instructions for determining whether rendering is necessary, for execution on a data processing device, e.g. the proxy server itself or a data processing device connected to the proxy server. The retrieving means may thus be constituted by a programmed data processing unit or be realized in hardware.

[0083] Rendering may for example be needed in case the client does not have the necessary software tools to visualize the requested data. Information on software tools available at the client may for example be transmitted from the client to the proxy server with the request for data or may be provided with client specific information available at the proxy server. For example, client specific information specifying browser capabilities or software tools available at the client could be stored beforehand in a client file in a memory which is accessible from the proxy server. The client specific information could be generated at the time when the client logs onto the proxy server and upon receiving a request the proxy

server could retrieve and analyse the client file and a decision whether rendering is required could be based on the analysing result. The client specific information could also include user preferences configured beforehand at the client, specifying cases where rendering is desired.

[0084] Further, rendering may also be required in case it is determined that the requested data should not be directly handled by the client itself, e.g., in dependence on a file size of the requested data, as large file sizes introduce high latency, particularly in case the client is connected to the network through a low bandwidth communication link and/or in case the client has a low capacity central processing unit. For example, rendering could be decided to be necessary in case the client is communicating through a low bandwidth communication link such as a standard telephone line or a wireless link, and requests concerning documents with a file size larger than, e.g. 1 Mbyte are received at the proxy server.

[0085] Further, rendering could be decided to be necessary in case a request has high processing demand and the client is constituted, e.g. by a small size mobile device such as a mobile phone or a mobile organizer with limited processing capabilities.

[0086] Rendering may also be required in dependence on an application associated with the requested data or the type of operation to be performed with the requested data, e.g. in case a small part of a bitmap image is to be visualized at the client or in case only selected pages of a large document are to be displayed at the client, e.g. for scrolling through a document or for editing a part of a document.

[0087] It is noted that it is also possible that information whether rendering is desired is transmitted with the request for data from the client, e.g. contained in an URL. For example, it may be determined with request whether a document should be opened locally, i.e. without rendering, or whether the document should be rendered, i.e. handled remotely, and only data frames for display should be transmitted to the client. In this case the determining means 422 analyses the request from the client in order to determine whether rendering is necessary.

[0088] It is further noted that the decision on rendering may in some cases be reached without retrieving the requested data or only part of the requested data by the proxy server, e.g. based on the request from the client or header information of the requested data.

[0089] In case the determining means 422 concludes that the request received from the client 41 does not require any rendering steps, i.e., further processing or re-writing of data, the proxy server 42 may directly transmit the requested data to the client 41. However, it is noted that also in this case the proxy server may perform operations on the requested data, e.g., format conversions or similar.

[0090] In case it is determined that the requested data

requested data, e.g. at the proxy server or the client, it is possible that the requested data are not at all or not fully retrieved by the proxy server, but only information on the location of the requested data is transmitted to the processing server 15, for example a URL, as indicated by an arrow 117. The processing server can then retrieve the requested data from the data server in preparation of rendering, as indicated by an arrow 118.

[0103] In this case the proxy server may retrieve some of the requested data, for example a part of the requested data including data type information until a decision on rendering is possible and then stop retrieving the requested data. The already retrieved portion of the data may then be discarded. The temporary memory means 14 is not necessary in this case.

[0104] In the following a further embodiment of the invention will be described with respect to Fig. 5. Fig. 5 shows a schematic block diagram of an access system for accessing data in a network according to another embodiment of the invention.

[0105] The embodiment shown in Fig. 5 is similar to the embodiment shown with respect to Fig. 4, however, the requested data, after being intermediately stored in the temporary memory means, are rendered and transmitted to the client only upon request from the client.

[0106] Fig. 5 illustrates a client 51 including a data handler 511 and pre-selection means 512, a proxy server 42, the data server 13, the temporary memory means 14 and the processing server 43.

[0107] In the scenario shown in Fig. 5 the client, upon emitting a request for data to the proxy server 42 as illustrated by arrow 111, e.g. containing an URL, the proxy server 42 generates a dummy response or link message 521 for instructing the client to redirect the request for data to the processing server 43. This may be accomplished by a code section containing instructions executed at the proxy server or by dedicated hardware.

[0108] The link message may include information on the storage location of the requested data, e.g. an URL or similar, specifying the storage location in the temporary memory means 14. Further, the link message may include address information of the site responsible for further processing or rendering of the requested data, e.g., of the processing server 43, and may include information needed for establishing a communication link to the processing server 43. Address information of the site responsible for rendering of the requested data is particularly useful, in case a plurality of processing servers is provided.

[0109] However, it is also possible that the client may be configured to connect to a predetermined site upon receiving a link message, e.g. to the processing server 43, in which case the link message does not need to contain address information of the site responsible for rendering of the requested data.

[0110] The link message may further include information on required rendering steps and may include information on the data type or format of the requested data

after rendering, particularly in case the data format will change during rendering, e.g., a MIME type.

[0111] The link message may also include user information, e.g. a password for authentication purposes and/or a session identifier (session ID). A session ID may be particularly useful in case the client previously generated a similar request for data and the processing server already launched a required application for rendering beforehand and/or already accessed the requested data.

[0112] The link message will be transmitted from the proxy server 42 to the client 51 as illustrated by an arrow 521. Accordingly, instead of an expected response, e.g., containing requested data, the client receives from the proxy server a message indicating that the data may not be obtained from the proxy server 42 but from another site.

[0113] The client 51, upon receiving the link message from the proxy server 42, preferably analyses the link message and generates a corresponding request for the processing server 43 concerning the requested data and transmits same to the processing server as illustrated by an arrow 522.

[0114] This request may specify the expected data, e.g. an URL, and/or data type information and/or information on execution of a particular protocol at the client.

[0115] The processing server 43 in response to receiving the request from the client 51 preferably accesses the temporary memory means 14 and loads the originally requested temporarily stored data. Then, after performing the required rendering steps, as outlined before in detail with respect to previous embodiments, the processing server may transmit the rendered data to the client 51 for further handling and/or visualization purposes.

[0116] The client 51 may further comprise the data handler 511 and upon receiving the dummy response from the proxy server 42, may activate the data handler 511 for establishing a communication link between the client 51 and the processing server 43 and/or for handling data from the processing server 43.

[0117] The data handler 511 may be realized by a hardware unit or by executing a code section containing instructions on a data processing device for establishing a communication link between the client and the processing server and to generate commands for execution by the processing servers to manipulate the requested data and/or to visualize or further process received information. The data handler 511 may include e.g. a suitable plug-in such as a suitable plug-in, tools for page by page viewing, Active-X control, Java applet or similar.

[0118] For example, the data handler 511 may be started upon receiving the link message from the proxy server and further, the link message may include information for allowing the data handler to contact the processing server 43 for establishing the communication link. When the communication link is established,

requested data, transmission bandwidth, processing bandwidth and similar, as outlined before.

[0135] In case rendering is determined not to be necessary, i.e. the decision in step S62 is "NO", the data may be directly transmitted to the client and the flow ends.

[0136] In case the proxy server determines that rendering is necessary, i.e. the decision in step S62 is "YES", in a step S63 the proxy server generates a link message and transmits the link message to the client. As outlined before, the link message may include address information of the requested data indicating a storage location in the temporary memory means, such as an URL. Further, the link message may specify a host for performing the rendering, preferably the processing server. Furthermore, the link message may contain information allowing the client to establish a communication link to the processing server and finally, the link message may contain a user identifier (user ID) and/or a session identifier (session ID) specifying a communication link to the processing server.

[0137] In a step S64 the client receives and analyses the link message from the proxy server. In case the link message contains information on a protocol to be started in order to communicate with the processing server, corresponding software tools may be started at the client in order to contact the processing server. After establishing a communication link between the client and the processing server 15 the client in a step S66 transmits instructions regarding the requested data to the processing server.

[0138] In a step S67 the processing server retrieves the requested data and renders the data, preferably according to the requests received from the client and in a step S68 the processing server transmits the rendered data to the client.

[0139] In a step S69 the processing server determines whether the client started an interactive session, e.g. whether the client wishes to control an application program running at the client.

[0140] In case the client started an interactive session the flow returns to step S66 wherein the client transmits further instructions regarding the requested data. For example, in case the session concerns the visualization of a document including scrolling, further instructions from the client could relate to scrolling through the document. Further, in case parts of a document should be edited, further instructions could specify parts of a document to be edited. In this case, after editing the edited part of the document would be transmitted back from the client to the processing server and introduced into the document.

[0141] In case the client and processing server are not involved in an interactive session, the flow of processing steps ends.

[0142] It is noted that all processing steps described with respect to Fig. 6 may be realized by a program or code sections executed on a system of data processing

devices.

[0143] In the following a further embodiment of the invention will be described with respect to Fig. 7. Fig. 7 shows a flow of messages transmitted between the different entities of an access system according to an embodiment of the invention.

[0144] In Fig. 7 messages are transmitted between the client 11, for example a client application or browser, the proxy server 12, the data server 13, the processing server 15 and the temporary memory means 14. The vertical lines indicate evolving time t in downward direction at each entity of the system.

[0145] In a step S701 a request for data is generated at the client 11 and is transmitted to the proxy server 12.

This request may, for example, include an URL concerning a "StarWriter" (.sdw) document, a data format part of the "StarOffice" suite. The client may be involved in a direct communication with the proxy server or through a packet switched network, including wireless connections. In case of a packet switched network the request may include a URL which may be translated by a DNS (directory network server) into an IP-address (internet protocol address) and a port number, e.g. port 80 for a HTTP (hypertext transport protocol) connection for a HTML (hypertext markup language) document.

[0146] The proxy server is preferably registered at the client as proxy, and therefore requests generated at the client will be sent to the proxy server for further handling.

[0147] Upon receiving the request from the client 11, the proxy server 12 forwards the request in a step S702 to the data server 13, preferably including his own network address. Since, as for example common in network applications with packet transmission, a request may preferably include origination address and destination address, upon receiving the request transmitted in step S702 to the data server, the data server retrieves the requested data, e.g. from its memory and transmits the data back to the proxy server 12 in a step S703.

[0148] After receiving the requested data, the proxy server 12 determines whether the retrieved data need to be rendered, i.e. further processed or rewritten, before being transmitted to the client for further handling including visualization, as it was outlined with respect to the previous embodiments

[0149] The step of determining whether rendering is necessary may be executed at the proxy server 12 before or after receiving the requested data. In case it is not possible to determine from the request itself, as transmitted from the client 11, whether the requested data need to be rendered, the requested data will preferably be retrieved first.

[0150] In case it is determined that the data need to be rendered, the requested data are transmitted in a step S704 from the proxy server 12 to the temporary memory means 14 for temporary storage.

[0151] In case it is determined that the requested data do not have to be rendered or further processed, as the client may or should itself handle the data, they may di-

[0166] In the following a further embodiment of the invention will be described with respect to Fig. 8. Fig. 8, similar to Fig. 7, shows a time sequence of events occurring at the client 11, the pre-selection means 512, the proxy server 12 and the data server 13.

[0167] In a step S801 a request for data is transmitted from the client to the pre-selection means. The request for data from the client corresponds to the request for data transmitted in step S701 in Fig. 7, with the only difference that it is not transmitted to the proxy server, but to the pre-selection means. The pre-selection means 512 may be realized as a hardware unit or as a code section containing instructions for execution at a data processing device. The pre-selection means 512 may be located at the client or may be located remote to the client and performs a pre-selection of requests for data into requests requiring rendering of data and requests not requiring rendering of data and for transmitting requests requiring rendering to the proxy server and directly retrieves data in case it is determined that the requested data do not require rendering before transmission to the client.

[0168] The pre-selection may be based on the data type of the requested data and thus may be obtained from the request itself or may be determined based on further heuristics. It is also possible that the pre-selection depends on a domain name included into the request, as outlined before.

[0169] In case it is determined at the pre-selection means that a request requiring rendering is present, this request is transmitted from the pre-selection means 512 to the proxy server 12 in a step S802.

[0170] In this embodiment the proxy server is preferably registered at the client, i.e., at the pre-selection means as a proxy for all requests determined at the pre-selection means to require rendering, all other requests are directly executed.

[0171] The proxy server forwards the request to the data server 13 in step S803 and which transmits the requested data in step S804 to the proxy server 12. These steps correspond to steps S702 and S703 described with respect to Fig. 2.

[0172] Thereafter the proxy server confirms whether rendering is indeed required, based on the retrieved data, and stores the data in the temporary memory means or transmits the data back to the client, depending on the determination result, as for example described with respect to previous embodiments. The rendered data may be transmitted from the processing server 15 to the client 11 directly or may be transmitted from the processing server through the pre-selection means 512 to the client 11.

[0173] The subsequent steps executed for rendering and transmitting the rendered data to the client correspond to the further steps previously described with respect to Fig. 7.

[0174] However, if it is determined at the pre-selection means 512 that a request for data was received from

the client 11 which does not require rendering of data, a direct retrieval message is sent from the pre-selection means to the data server 13 in step S810, e.g. via a packet-switched network. The data server 13 upon receiving the message from the pre-selection means transmits the data either directly to the client 11 or through the pre-selection means.

[0175] In the following a further embodiment of the invention will be described with respect to Fig. 9. Fig. 9 shows a schematic block diagram of functional entities of an access system according to an embodiment the invention.

[0176] A first client denoted with reference numeral 111 is shown to be part of public network 19. A second client denoted with reference numeral 112 is shown as part of the local area network 18. The first client 111 may be connected to the local area network via the wide area network or directly via a dedicated communication line such as a telephone line. Data transmission may be encrypted.

[0177] Further, Fig. 9 shows three processing servers 151, 152 and 153, wherein processing servers 151 and 152 are part of the local area network 18 and wherein the third processing server 153 is part of the public network.

[0178] Still further, Fig. 9 shows four data servers 131, 132, 133 and 134, wherein the data servers 131, 132 and 133 are part of the wide area network 19 and the fourth data server 134 is part of the local area network 18. Of course, an arbitrary number of clients, proxy servers, processing servers and data servers may be provided inside the local area network and/or in the wide area network.

[0179] In case a request for data is generated by the first client 111, it will be transmitted either via a dedicated line or the wide area network 19, e.g., including wireless transmission, to the local area network and to the proxy server 12.

[0180] In case a request for data is generated at the second client 112, which is part of the local area network 18, it will be transmitted through the local area network to the proxy server 12.

[0181] Upon receiving the request from any of the shown clients, the proxy server 12 will retrieve the requested data from an appropriate one of the data servers 131, 132, 133 and 134. In case the corresponding data server is part of the local area network, the retrieval of the data by the proxy server may be executed through the local area network, in case the data server is part of the wide area network 19, the requested data will be transmitted from the wide area network to the local area network. This transmission may cross a firewall (not shown), protecting the local area network from unauthorized access from the outside.

[0182] A firewall generally is a method for keeping a network secure. It can for example be implemented in a router that filters out unwanted packets, or it may use a combination of technologies in routers and hosts. Fire-

containing instructions for establishing a communication link between the client and the processing server and for receiving the rendered data from the processing server 15; 151, 152, 153; 43.

8) Access system according to 1), wherein the proxy server 12; 42 includes a code section containing instructions for direct transmission of the requested data to the client in case it is determined by determining means 422 that the requested data do not have to be rendered before transmission to the client.

9) Access system according to 1), wherein the client 11; 111, 112; 41; 51 includes a code section containing instructions for performing a pre-selection of requests for data into requests requiring rendering of data and requests which do not require rendering of data and for transmitting requests requiring rendering to the proxy server 12; 42, and for directly retrieving data in case it is determined that the requested data do not require rendering before transmission to the client.

10) Access system according to 2) or 3), wherein at least the proxy server 12; 42, the processing server and the temporary memory means are connected by a local network 18.

11) Access system according to 1), comprising

a plurality of processing servers;

a code section containing instructions to determine at least one of the processing servers for rendering; and

wherein the proxy server 12; 42 includes a code section containing instructions for receiving requests for data from a plurality of clients and for retrieving data from a plurality of data servers 13; 131, 132, 133.

12) Access system according to 1), wherein the processing server 15; 151, 152, 153; 43 includes a code section containing instructions for transmitting the rendered data to the client 11; 111, 112; 41; 51 on a return path including the proxy server 12; 42.

13) Access system according to 1), wherein the processing server and the proxy server are constituted by a single data processing device.

14) A client may be adapted to cooperate with the access system according to 1), comprising

a code section containing instructions to generate a request for data,

a code section containing instructions to receive and analyse a link message from the access system; and

a code section containing instructions for establishing a communication link between the client and the processing server and for receiving rendered data from the processing server.

15) The client according to 14) may comprise a code section containing instructions for performing a pre-selection of requests for data into requests requiring rendering of data and requests which do not require rendering of data and for transmitting requests requiring rendering to the proxy server 12; 42, and for directly retrieving data at the client in case it is determined that the requested data do not require rendering before transmission to the client. According to another embodiment of the invention, a proxy server of the access system may be constituted as follows:

16) Proxy server for accessing data in a network, comprising

a code section containing instructions for receiving a request for data from a client 11; 111, 112; 41; 51,

a code section containing instructions for determining whether the requested data have to be rendered before transmission to the client, and

a code section containing instructions for authorizing a processing server 15; 151, 152, 153; 43 for rendering the data and for transmitting the rendered data to the client 11; 111, 112; 41; 51.

17) Proxy server according to 16) comprising

a code section containing instructions for retrieving the requested data from a data server 13; 131, 132, 133, 134,

a code section containing instructions for storing the requested data in temporary memory means 14 if it is determined that the requested data have to be rendered before transmission to the client, and

a code section containing instructions for authorizing the processing server 15; 151, 152, 153; 43 to retrieve the data stored in the temporary memory means.

18) Proxy server according to 16), wherein the proxy server includes a code section containing in-

- the processing server (15; 151, 152, 153; 43),
and
- the processing server is arranged to retrieve
the requested data from a data server (13; 131,
132, 133, 134) using the address information.
4. Access system according to claim 2 or 3, wherein
the proxy server (12; 42) includes
- means for generating a link message contain-
ing the address information of the requested
data; and
- means for transmitting the link message to the
client (11; 111, 112; 41; 51).
5. Access system according to claim 4, wherein the
link message further includes at least one of the
group consisting of
- address information of the processing server,
- data type information describing the requested
data,
- a client identifier, and
- a session identifier.
6. Access system according to one of the claims 2 - 5,
wherein the address information of the requested
data is constituted by an URL and the data type in-
formation by a MIME type.
7. Access system according to one of the claims 4 - 6,
comprising at the client a data handler (511) for es-
tablishing a communication link between the client
and the processing server and for receiving the ren-
dered data from the processing server (15; 151,
152, 153; 43).
8. Access system according to one of the preceding
claims, wherein the proxy server (12; 42) includes
means for direct transmission of the requested data
to the client in case it is determined by determining
means (422) that the requested data do not have to
be rendered before transmission to the client.
9. Access system according to one of the preceding
claims, comprising pre-selection means (512) lo-
cated at the client (11; 111, 112; 41; 51) for perform-
ing a pre-selection of requests for data into requests
requiring rendering of data and requests which do
not require rendering of data and for transmitting re-
quests requiring rendering to the proxy server (12;
42), and for directly retrieving data in case it is de-
termined that the requested data do not require ren-
- dering before transmission to the client.
10. Access system according to one of the claims 2 - 9,
wherein at least the proxy server (12; 42), the
processing server and the temporary memory
means are connected by a local network (18).
11. Access system according to one of the preceding
claims, comprising
- a plurality of processing servers;
- means to determine at least one of the process-
ing servers for rendering; and
- wherein the proxy server (12; 42) is arranged
to receive requests for data from a plurality of clients
and to retrieve data from a plurality of data servers
(13; 131, 132, 133).
12. Access system according to one of the preceding
claims, wherein the processing server (15; 151,
152, 153; 43) is arranged to transmit the rendered
data to the client (11; 111, 112; 41; 51) on a return
path including the proxy server (12; 42).
13. Access system according to one of the preceding
claims, wherein the processing server and the
proxy server are constituted by a single data
processing device.
14. Client adapted to cooperate with the access system
according to any of the preceding claims, compris-
ing
- means to generate a request for data,
- means to receive and analyse a link message
from the access system; and
- a data handler (511) for establishing a commu-
nication link between the client and the
processing server and for receiving rendered
data from the processing server.
15. Client according to claim 14, comprising pre-selec-
tion means (512) for performing a pre-selection of
requests for data into requests requiring rendering
of data and requests which do not require rendering
of data and for transmitting requests requiring ren-
dering to the proxy server (12; 42), and for directly
retrieving data at the client in case it is determined
that the requested data do not require rendering be-
fore transmission to the client.
16. Proxy server for accessing data in a network, com-
prising

a data server (13; 131, 132, 133, 134).

27. Processing server according to claim 23 or 26, comprising receiving means for receiving an instruction message from a data handler (511) at the client to establish a communication link between the client and the processing server (15; 151, 152, 153; 43).

28. Client adapted to cooperate with the processing server according to claims 25 - 27, comprising

means to generate a request for data,

means to receive and analyse a link message from the access system; and

a data handler (511) for establishing a communication link between the client and the processing server and for receiving rendered data from the processing server.

29. Client according to claim 28, comprising pre-selection means (512) for performing a pre-selection of requests for data into requests requiring rendering of data and requests which do not require rendering of data and for transmitting requests requiring rendering to the proxy server (12; 42), and for directly retrieving data at the client in case it is determined that the requested data do not require rendering before transmission to the client.

30. Method for accessing data in a network, comprising:

receiving a request for data from a client (11; 111, 112; 41; 51) at a proxy server (12; 42);

determining whether the requested data have to be rendered before transmission to the client;

rendering the data at a processing server (15; 151, 152, 153; 43); and

transmitting the rendered data to the client (11; 111, 112; 41; 51).

31. Method according to claim 30, comprising

retrieving the requested data from a data server (13; 131, 132, 133, 134);

storing the requested data in temporary memory means (14) if it is determined at the proxy server (12; 42) that the requested data have to be rendered before transmission to the client; and

retrieving the data stored in the temporary

memory means by the processing server (15; 151, 152, 153; 43).

32. Method according to claim 30, wherein

the proxy server (12; 42) transmits address information of the requested data to the processing server (15; 151, 152, 153; 43), and

the processing server retrieves the requested data from a data server (13; 131, 132, 133, 134) using the address information.

33. Method according to claim 30 - 32, comprising

generating a link message containing address information of the requested data; and

transmitting the link message to the client (11; 111, 112; 41; 51).

34. Method according to claim 33, wherein the link message further includes at least one of the group consisting of

address information of the processing server,

data type information describing the requested data,

a client identifier, and

a session identifier.

35. Method according to one of the claims 30 - 34, wherein the address information of the requested data is constituted by an URL and the data type information is a MIME type.

36. Method according to one of the claims 30 - 35, comprising activating a data handler (511) for establishing a communication link between the client and the processing server and for receiving the rendered data from the processing server (15; 151, 152, 153; 43).

37. Method according to one of the claims 30 - 36, comprising directly transmitting the requested data to the client in case it is determined by the determining means (422) that the requested data do not have to be rendered before transmission to the client.

38. Method according to one of the claims 30 - 37, comprising pre-selecting requests for data into requests requiring rendering of data and requests not requiring rendering of data and for transmitting requests requiring rendering to the proxy server (12; 42), and for directly retrieving data in case it is determined

rendering the data at the processing server;
and

transmitting the rendered data to the client.

5

51. Method according to claim 50, wherein the processing server is instructed by a data handler (511) running at the client to retrieve the requested data from the temporary memory means (14).

10

52. Method according to claim 50, wherein

the processing server receives address information of the requested data, and

15

the processing server retrieves the requested data from a data server (13; 131, 132, 133, 134) using the address information.

53. A computer readable medium, having a program recorded thereon, where the program is to make the computer execute the method according to one of the claims 30 - 52.

20

54. A computer program product comprising the computer readable medium according to claim 53.

25

30

35

40

45

50

55

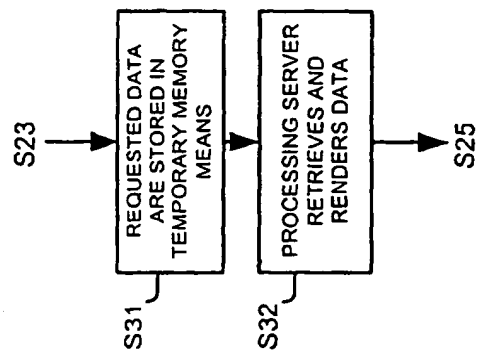


Fig. 3

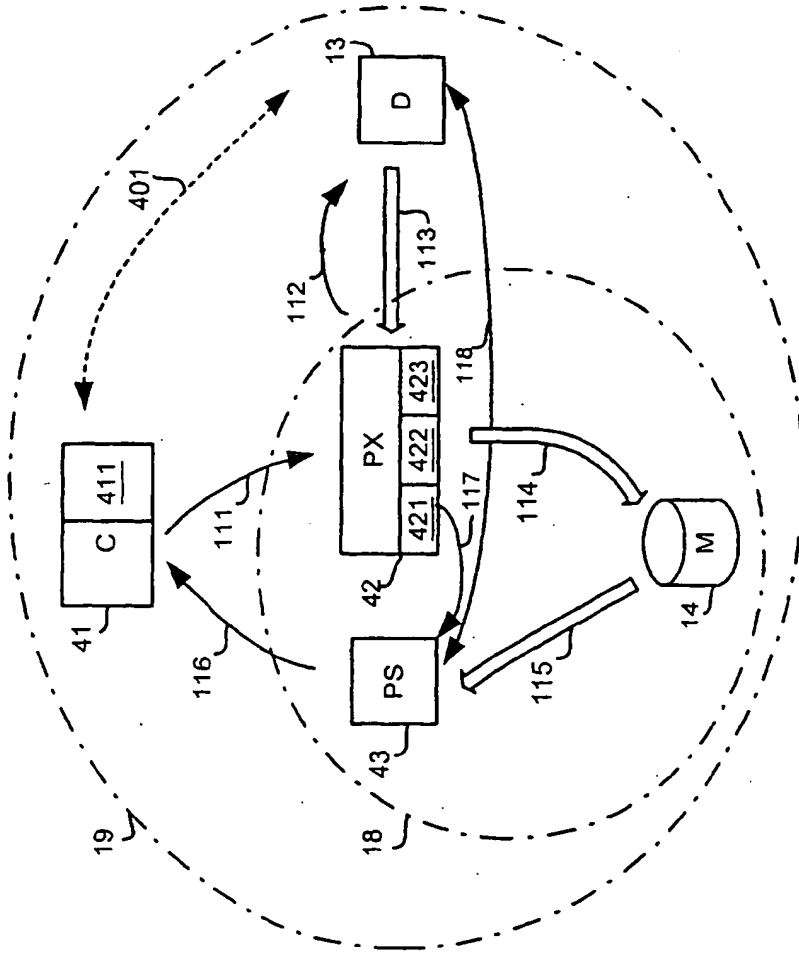


Fig. 4

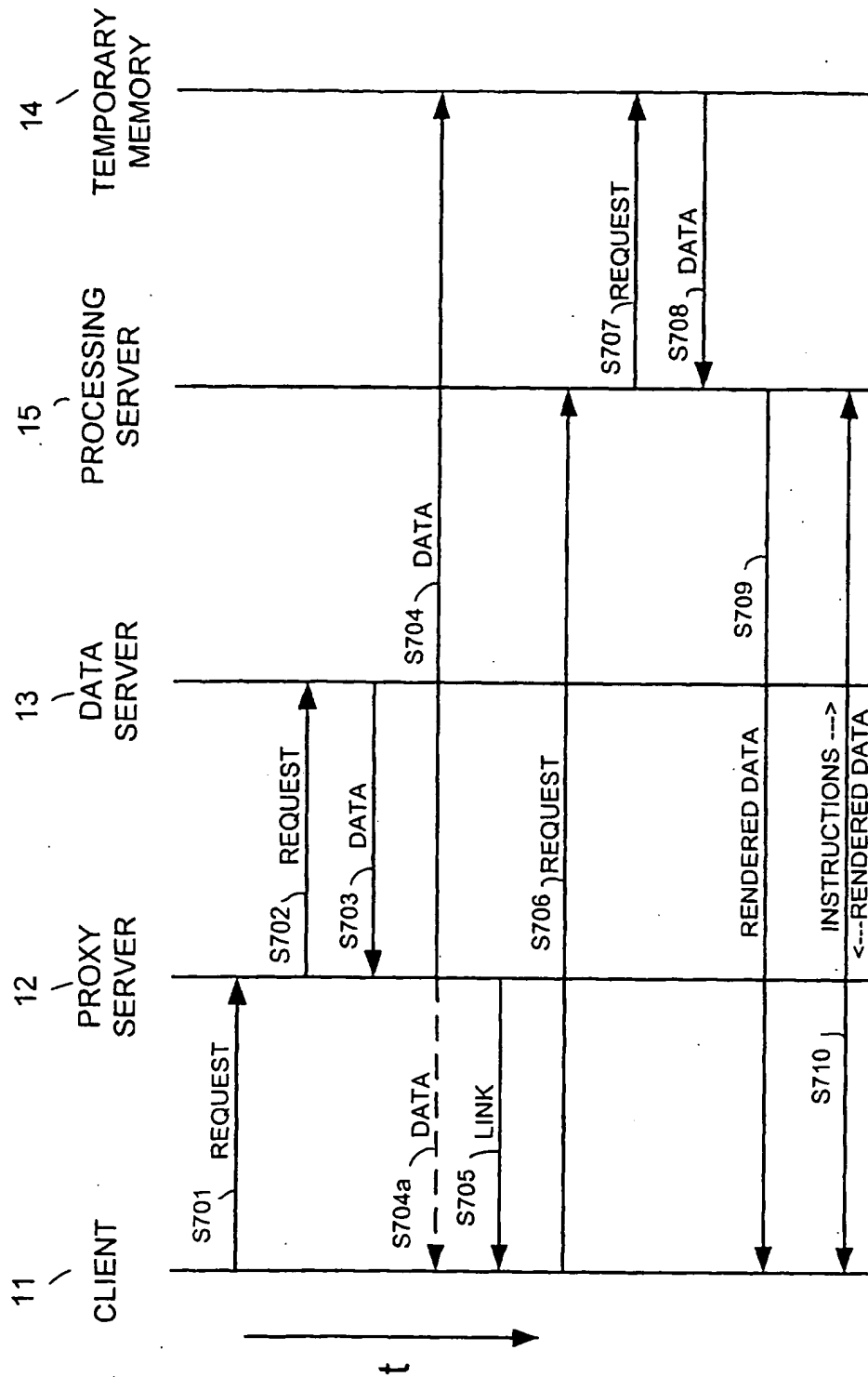


Fig. 7

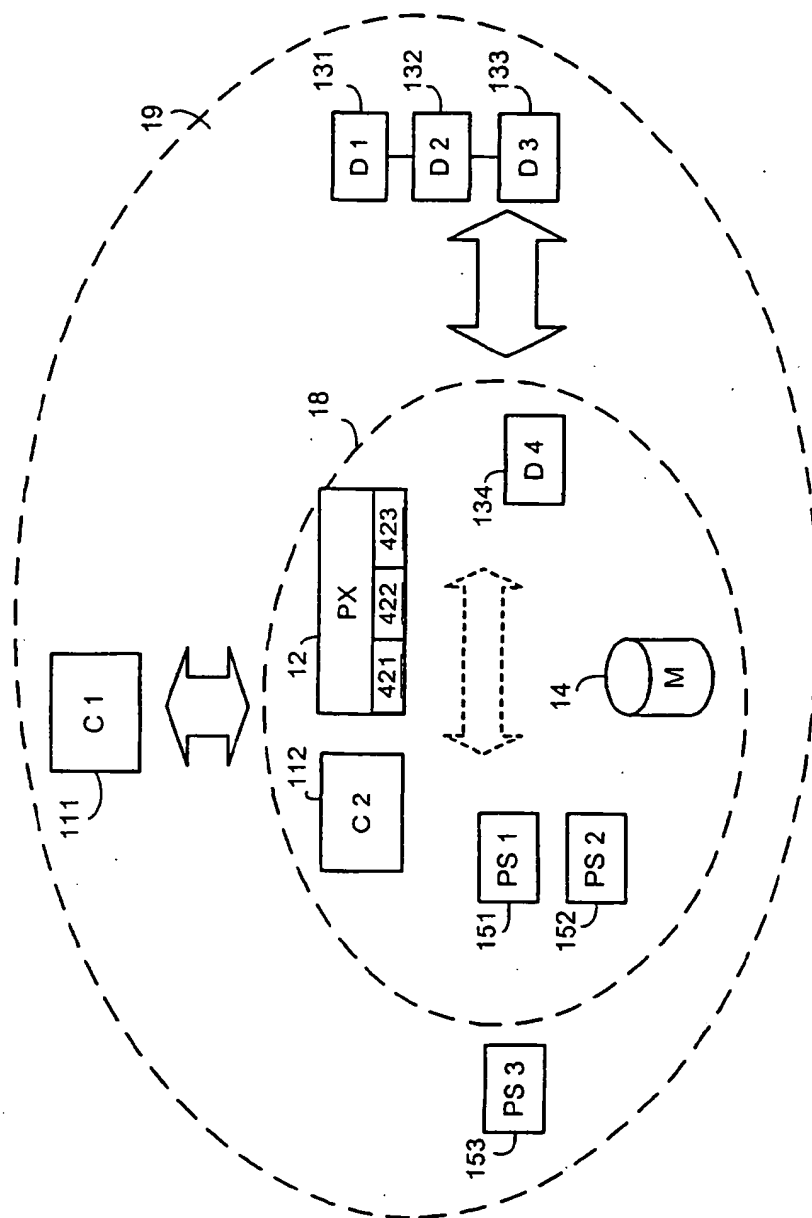


Fig. 9



DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
Y	FREYTAG C ET AL: "Resource adaptive WWW access for mobile applications" COMPUTERS AND GRAPHICS,GB,PERGAMON PRESS LTD. OXFORD, vol. 23, no. 6, December 1999 (1999-12), pages 841-848, XP004187832 ISSN: 0097-8493	14,15, 23,24, 28,29	
A	 * abstract * * page 842, left-hand column, line 20 - page 842, right-hand column, line 30 * * page 843, left-hand column, line 9 - page 844, left-hand column, line 20 * * page 846, left-hand column, line 7 - page 846, left-hand column, line 16 * ---	1,16,30, 42,50,53	
A	BROOKS ET AL: "Application-Specific Proxy Servers as HTTP Stream Transducers" INTERNET CITATION,XX,XX, 1 December 1995 (1995-12-01), pages 1-9, XP002081461 * abstract * * page 1, line 10 - page 3, line 19 * * page 7, line 10 - page 7, line 48 * --- -/--	1-54	
The present search report has been drawn up for all claims			TECHNICAL FIELDS SEARCHED (Int.Cl.7)
Place of search THE HAGUE		Date of completion of the search 12 March 2001	Examiner Boyadzhiev, Y
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

EPC FORM 1503 03 82 (P04C01)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 00 11 7311

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

12-03-2001

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9843177 A	01-10-1998	US 5902846 A	11-05-1999
		AU 6865698 A	20-10-1998
		BR 9811457 A	19-09-2000
		EP 1012733 A	28-06-2000
<hr/>			

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.